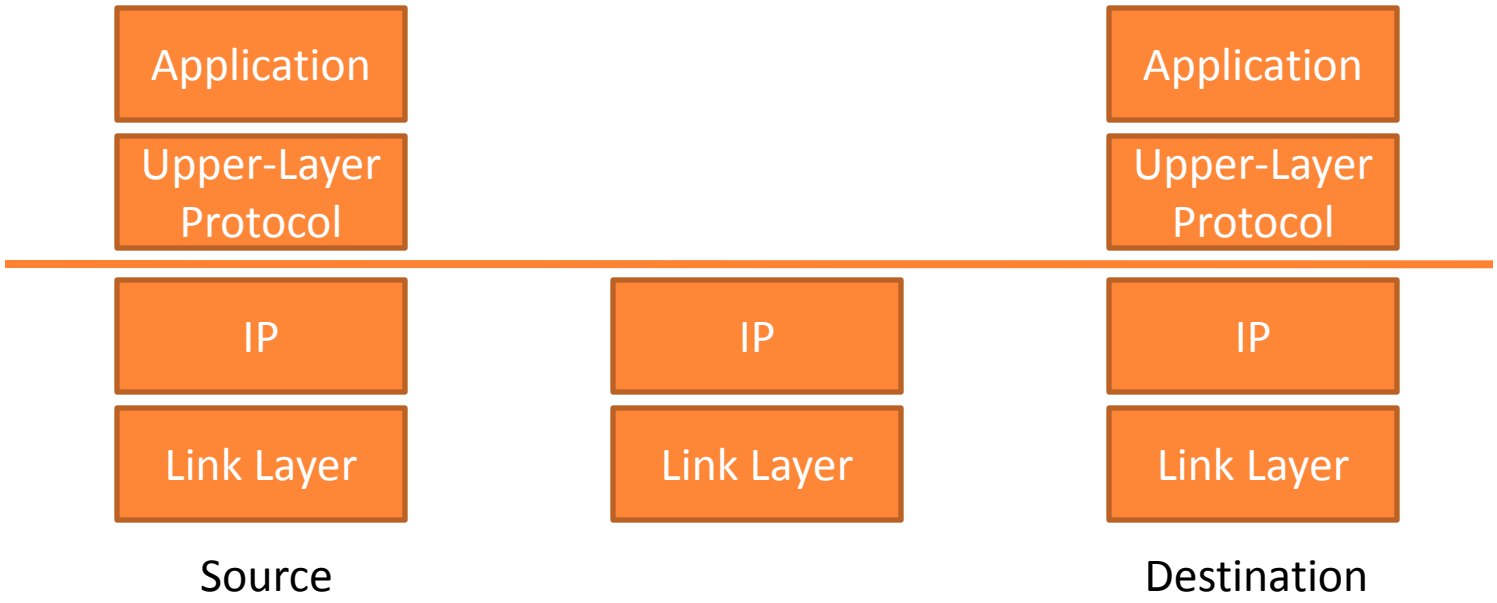
Evolution of the IP Model

draft-thaler-ip-model-evolution-01.txt

Dave Thaler

What is the IP Model?

- The model exposed by IP to higher layer protocols and applications



Evolution

- IP was published in a series of IENs starting in 1978, then RFC 760 in 1980 and finally RFC 791 in 1981
- However, the model continued to evolve.
- Since 1978 many applications and upper-layer protocols have evolved around various additional assumptions
  - They are not listed in one place
  - They are not necessarily well-known
  - They are not necessarily thought about when making changes
- Some changes are intentional, some changes happen as a side effect of some other goal
- By 1989, there was already some confusion and so RFC 1122 clarified many things and extended the model
- In 2004, RFC 3819 (“Advice for Internet Subnetwork Designers”) gave advice to L2 designers on things that affect upper layers
- (and various RFCs give advice on other specific topics: RFC 2991, 4903, etc)

Basic IP (RFC 791) Service Model

- Senders just send to an address, without signaling a priori
- Receivers just listen on an already provisioned address, without signaling a priori
- Packets can be of any size
- No guarantee of reliability
- No guarantee of ordering
- No guarantee of lack of duplication

End-system models (RFC 1122)

- Strong host:
  - Outgoing datagrams MUST be sent on the interface with the source address
  - Incoming datagrams MUST arrive on the interface with the destination address
- Weak host:
  - Outgoing datagrams can be sent out any interface
  - Incoming datagrams can arrive on any interface

- Note that enabling forwarding results in weak host
- Some OS's use strong host, some use weak host

IP Subnet Model

- Section 2.1 of [RFC4903] discusses the terms "link" and "subnet" with respect to the IP model.

- A "link" in the IP service model refers to the topological area within which a packet with IPv4 TTL or IPv6 Hop Limit of 1 can be delivered. That is, where no IP-layer forwarding (which entails a TTL/Hop Limit decrement) occurs between two nodes.

- A "subnet" in the IP service model refers to the topological area within which addresses from the same subnet prefix are assigned to interfaces.

But wait… there's more!

- Common application/upper-layer protocol assumptions (or myths, increasingly…)
  - Assumptions about routing
  - Assumptions about addressing
  - Assumptions about upper-layer protocol extensibility
  - Assumptions about security

Assumptions about routing

Connectivity is Symmetric

- Examples of behavior:
  - Apps do request-response, callbacks, etc
- Status:
  - Much less true with NAT, firewall, 802.11 ad-hoc, satellite, admission control proxies, etc.
  - UDLR was one effort to help restore
  - Request-response *usually* works, but not callbacks

Connectivity is Transitive

- Examples of behavior:
  - Apps do referrals/redirects
- Status:
  - Much less true with NAT, firewall, 802.11 ad-hoc, satellite, etc.

Broadcast/multicast is supported within a link

- Examples of behavior:
  - Service discovery
  - DHCP, ARP, etc
- Status:
  - Various NBMA links exist, including X.25, ATM, frame relay, 6to4, ISATAP, Teredo
  - Some recent semi-broadcast links: 802.11 ad hoc, MANET

Broadcast/multicast is less expensive than replicated unicast

- Example of behavior:
  - Protocols use bcast/mcast over a link even if the destinations are all known

- Status:
  - Not true on many wireless links today (must send at lowest common denominator)
  - Generally true on wired today, but that could change

Assumptions about addressing

Addresses are stable over long periods of time

- Examples of behavior:
  - Apps resolve names to addresses and cache them without any notion of lifetime
  - Name resolution APIs don't even provide the lifetime
- Status:
  - Much less true with DHCP, roaming, etc.
  - PMIP trying to restore within a local network
  - MIP, HIP, etc trying to restore to some extent by adding an additional address that is stable

A host has only one address and one interface

- Examples of behavior:
  - Apps resolve name to address and just use the first one returned
  - Some apps use address to identify users/machines
  - Some DHCP options are defined as machine-wide
- Status:
  - Much less true with multihoming, dual-stack nodes, VPNs, etc.
  - MIP, HIP, etc trying to restore to some extent

Selecting a local address selects the interface

- Example of behavior:
  - App binds to a specific address and expects to only get traffic on that interface
- Status:
  - Not true if forwarding is enabled, or host follows “weak end-system” model

Many more

- Presented in EXPLISP
  - E2E delay of first packet to a destination is typical
  - Reordering is rare
  - Loss is rare and probabilistic, not deterministic
  - An address used by an app is the same as the addr used for routing
- 8 more in draft

Impact

- Any changes to assumptions break some apps
  - Ossification of the Internet means changes cause pain
  - Changes must be done with extreme care
- Adding *optional* functionality is generally safe
  - But fewer apps use

- Draft collects assumptions in one place, but need to consider them when making changes at network layer or below